PARSING A FLEXIBLE WORD ORDER LANGUAGE

Vladimir Pericliev and Alexander Grigorov
Institute of Mathematics,
Acd. G. Bonchev Str., bl.8, 1113 Sofia, Bulgaria,
E-mail: peri@bgearn.bitnet and grigorov@bgearn.bitnet

Abstract

A logic formalism is presented which increases the expressive power of the ID/LP format of GPSG by enlarging the inventory of ordering relations and extending the domain of their application to non-siblings. This allows a concise, modular and declarative statement of intricate word order regularities.

1 Introduction

Natural languages exhibit significant word order (WO) variation and intricate ordering rules. Despite the fact that specific languages show less variation and complexity in such rules (e. g. those characterized by either fixed, or totally free, WO), the vast majority of world languages lie somewhere in-between these two extremes (e. g. *Steele 1981*). Importantly, even the proclaimed examples of rigid WO languages (English) exhibit variation, whereas those with proclaimed total scrambling (Warlpiri; cf. *Hale 1981*) show restrictions (*Kashket 1987*). Therefore, we need general grammar formalism, capable of processing "flexible" WO (i.e. complex WO regularities, including both extremes).

There seem to be a number of requirements that such a formalism should (try to) fulfil (e. g. *Pericliev and Grigorov 1992*). Among these stand out the formalism's:

(i) *Expressive power*, i. e. capability of (reasonably) handling complex WO phenomena, or "flexible" WO.

(ii) *Linguistic felicity*, i. e. capability of stating concisely and declaratively WO rules in a way maximally approximating linguistic parlance in similar situations.

(iii) *Modularity*, i. e. the separation of constituency rules from the rules pertaining to the linearization of these constituents (for there may be many, and diverse, reasons for wanting linearization (and constituency) rules easily modifiable, incl. the transparency of WO statements, the imprecision of our current knowledge of ordering rules or the wish to tailor a system to a domain with specific WO).

(iv) *Reversibility*, i. e. the ability of a system to be used for both parsing and generation (the reason being that, even if the system is originally intended for a parser, complex WO rules may be conveniently tested in the generation mode; in this sense it is not incidental that e. g. *Kay & Karttunen 1984* have first constructed a generator, and used it as a tool in testing the (WO) rules of their grammar, and only then have converted it into a parser).

In this paper, we present a logic-based formalism which attempts to satisfy the above requirements. A review shows that most previous approaches to WO within the logic grammars paradigm (*Dahl & Abramson 1990*) have not been satisfactory. Definite Clause Grammar, DCG, (*Pereira & Warren 1980*), with their CF-style rules, are not modular (in the sense above), so will have to specify explicitly each ordering of constituents in a separate rule, which results in an intolerably great number of rules in parsing a free WO language (e. g. for 5 constituents, which may freely permute, the number of rules is 5! = 120). Other approaches center around the notion of a "gap" (or "skip"). In Gapping Grammar (GG), for instance (*Dahl & Abramson 1984*, esp. Dahl 1984), where a rule with a gap may be viewed as a meta-rule, standing for a set of CF rules, free WO is more economically expressed, however, due the unnaturalness of expressing permutations by gaps, GGs generally are clumsy for expressing flexible WO, WO is not declaratively and modularly expressed, and GGs cannot be used for generation (being besides not efficiently implementable). Another powerful formalism, Contextual Discontinuous Grammar (*Saint-Dizier 1988*), which overcomes the GGs problems with generative capacity, is also far from being transparent and declarative in expressing WO (e. g. rules with fixed WO are transformed into free order ones by introducing special rules, containing symbols with no linguistic motivation, etc.).

2 Problems for the ID/LP format

In the Immediate Dominance/Linear Precedence (ID/LP) format of GPSG (*Gazdar & Pullum 1981*, *Gazdar et al. 1985*), where the information, concerning constituency (= immediate dominance) and linear order, is separated, WO rules are concisely, declaratively and modularly expressed over the domain of local-trees (i. e. trees of depth 1). E. g. the ID rule $\mathbf{A} \rightarrow_{ID} \mathbf{B, C, D}$, if no linearization restrictions are declared,stands for the mother node expanded into its siblings appearing in any order;declaring the restriction $\{ \mathbf{D} < \mathbf{C} \}$ e. g., it stands for the CFG rules $\{ \mathbf{A} \rightarrow \mathbf{B\ D\ C}, \mathbf{A} \rightarrow \mathbf{D\ B\ C}$ and $\mathbf{A} \rightarrow \mathbf{D\ C\ B} \}$.

It is important to note that in GPSG the linear precedence rules stated for a pair of sibling constituents should be valid for the whole set of grammar rules in which these constituents occur, and not just for some specific rule (this ”global” empirical constraint on WO is called the Exhaustive Constant Partial Ordering (ECPO) property).

However, there are problems with ECPO. They may be illustrated with a simple example from Bulgarian. Consider a grammar describing sentences with a reflexive verb and a reflexive particle (the NP-subject and the adverb being optional), responsible for expressions whose English equivalent is e. g. ”(Ivan) shaved himself (yesterday)”.

(1) $\mathbf{S} \rightarrow_{ID}$ **NP, VP**

(2) $\mathbf{S} \rightarrow_{ID}$ **VP** % omitted subject

(3) $\mathbf{VP} \rightarrow_{ID}$ **V[refl], Part[refl], Adv**

(4) $\mathbf{VP} \rightarrow_{ID}$ **V[refl], Part[refl]** % omitted adverb

First, assume we derive a sentence, applying rules (2) and (3). (5a-b) are the only acceptable linearizations of the sister constituents in (3).

(5a) Brasna (**V[refl]**) se (**Part[refl]**) vcera (**Adv**)
shaved himself yesterday

(5b) Vcera (**Adv**) se (**Part[refl]**) brasna (**V[refl]**)
Yesterday himself shaved
(meaning: (Someone) shaved himself yesterday)

LP rules however cannot enforce exactly these orderings because the CFG, corresponding to (5a-b), viz.

(6) $\mathbf{A} \rightarrow \mathbf{B\ C\ D}$
$\mathbf{A} \rightarrow \mathbf{D\ C\ B}$

is non-ECPO. Thus, fixing any ordering between any two constituents in (3) will, of necessity, block at least one of the correct orderings (5a-b); alternatively, sanctioning no WO restriction will result in overgeneration, admitting, besides the grammatical (5a-b), 4 ungrammatical permutations. This inability to impose an arbitrary ordering on siblings we will call the ordering-problem of ID/LP grammars.

Now assume we derive a sentence, applying rules (1) and (4). The ordering of the siblings, reflexive verb and particle, in (4) now depends on the order of nodes **NP** and **VP** higher up in the tree in rule (1): if **NP** precedes **VP** in (1), then the reflexive particle must precede the verb in (4), otherwise it should follow it.

(7a) Ivan (**NP**) se (**Part[refl]**) brasna (**V[refl]**)
Ivan himself shaved

(7b)	Brasna (**V**[**refl**])	se (**Part**[**refl**])	Ivan (**NP**)
	Shaved	himself	Ivan

(meaning: Ivan shaved himself)

Again we are in trouble since LP rules cannot impose orderings among non-siblings, their domain of application being just siblings. This we call the *domain-problem* of ID/LP grammars. It is essential to note that the domain-problem may not be remedied (even if we are inclined to sacrifice linguistic intuitions) by ”flattening” the tree, e. g. collapsing rules (1) and (4) into

(8) **S** $\rightarrow_{ID}$ **NP, V[refl], Part[refl]**

Escaping the second problem, thrusts us into the first: we now cannot properly order the siblings, the CFG, corresponding to (7a-b), being the non-ECPO (6).

Sporadic counter-evidence for ECPO grammars has been found for some languages like English (the verb-particle construction, *Sag 1987*, *Pollard and Sag 1987*), German (complex fronting, *Uszkoreit 1985*, *Engelkamp et al. 1992*) and Finnish (the adverb myos 'also, too' *Zwicky and Nevis 1986*). Bulgarian offers massive counter-evidence (*Pericliev 1992b*); one major example, the Bulgarian clitic system, we discuss in Section 4.

3 The formalism

EFOG (Extended Flexible word Order Grammar) extends the expressive power of the ID/LP format. First, EFOG introduces further WO restrictions in addition to precedence (enabling it to avoid the ordering-problem), and, second, the formalism extends the domain of application of these WO restrictions (in order to handle the domain-problem).

In the immediate dominance part of rules EFOG has two types of constituents: *non-contiguous* (notated: **#Node**) and contiguous (notated just: **Node**), where **Node** is some node. Informally, a contiguous node shows that its daughters form a contiguous sequence, whereas a non-contiguous one allows its daughters to be interspersed among the sisters of this non-contiguous node. E. g. in EFOG notation (using a double arrow for ID rules, small case letters for constants and upper case ones for variables), the grammar of the Latin sentence: *Puella bona puerum parvum amat (good girl loves small boy)*, grammatical in all its 120 permutations and, besides, having discontinuity in the noun phrases, we capture with the following structured EFOG rules with no WO restrictions:

s ⇒ #np(nom), #vp.
np(Case) ⇒ adj(Case), noun(Case).
vp ⇒ verb, #np(acc).

accompanied by the dictionary rules:

verb ⇒ [amat].
adj(nom) ⇒ [bona].
adj(acc) ⇒ [parvum].
noun(nom) ⇒ [puella].
noun(acc) ⇒ [puerum].

The non-contiguous nodes allow us to impose an ordering (or to intersperse, as in the above case) all their daughter nodes without having to sacrifice the natural constituencies. It will be clear that this extension of the domain of LP rules (which can go any depth we like), besides ordering between non-siblings, allows an elegant treatment of discontinuities.

In order to solve the ordering-problem, we have introduced additional WO constraints. The following atomic WO constraints have been defined:

- *Precedence constraints*:
  - precedes (e. g. **a** < **b**)

  - immediately precedes (**a** << **b**) (we also maintain the notation, > and >>, for (immediately) follows; see commentary below)

- *Adjacency constraints*:
  - is adjacent (**a** <> **b**).

- *Position constraints*:
  - is positioned first/last (e. g. **first(a, Node)**, where **Node** is a node; e. g. **first(a, s)** designates that **a** is sentence-initial.

We also allow atomic WO constraints to combine into complex logical expressions, using the following operators with obvious semantics:

- Conjunction (notated: **and**)
- Disjunction (**or**)
- Negation (**not**)
- Implication (**if**, e. g. **(b >> a) if (a < c)**)
- Equivalence (**iff**, e. g. **(b >> a) iff (a < c)**)
- Ifthenelse (**ifthenelse**)

Our WO restriction language is, of course, partly logically redundant (e. g. immediately precedence may be expressed through precedence and adjacency, and so is the case with the last two of the operators, etc.). However, what is logically is not necessarily psychologically equivalent, and our goal has been to maintain a linguist-friendly notation (cf. requirement (ii) of Section 1). To take just one example, we have 'after' in addition to 'before', since linguists normally speak of precedence of dependent with respect to head word, not vice versa, and hence will use both expressions in respective situations (surely it is not by chance that NLs also have both words).

As a simple example of the ordering possibilities of EFOG, consider the WO Universal 20 (of Greenberg and Hawkins) to the effect that NPs comprising dem(onstrative), num(eral), adj(ective) and noun can appear in that order, or in its mirror-image. We can write a "universal" rule enforcing adjacent permutations of all constituents as follows:

np ⇒ dem, num, adj, noun.
lp: dem <> num and num <> adj and adj <> noun.

4 Bulgarian clitics

Bulgarian clitics fall into different categories:

(1) nominals (short accusative pronouns: *me* "me", *te* "you", etc.; short dative pronouns: *mi* "to me", *ti* "to you", etc.);

(2) verbs (the present tense forms of "to be" *sam* "am", *si* "(you) are", etc.);

(3) adjectives (short possessive pronouns: *mi* "my", *ti* "your", etc.; short reflexive pronoun: *si* "one's own");

(4) particles (interrogative *li* "do", reflexive *se* "myself/yourself…", the negative *ne* "no(t)", etc.).

They have the distribution of the specific categories they belong to, but show diverse, and quite complex orderings, varying in accordance with the positions of their siblings/non-siblings as well as the position of other clitics appearing in the sentence. [1] In effect, their ordering as a rule cannot be correctly stated in the standard ID/LP format.

By way of illustration, below we present the EFOG version (simplified for expository reasons) of the grammar (1-4) from Section 2 to get the flavour of how we handle the problems mentioned there. The ID rules are as follows (note that the non-contiguous node **#vp** allows its daughters **v(refl)**, **part(refl)**, and **adv** to be ordered with respect to **np**):

(1') **s ⇒ np, #vp.**

(2') **s ⇒ vp.** % omitted subject

(3') **vp ⇒ v(refl), part(refl), adv.**

(4') **vp ⇒ v(refl), part(refl).** % omitted adverb
np ⇒ [ivan].
v(refl) ⇒ [brasna].
part(refl) ⇒ [se].
adv ⇒ [vcera].

The WO of **v(refl)** and **part(refl)** is as follows. First, the reflexive particle never occurs sentence-initially (information we cannot express in ID/LP); in EFOG we express this as:

lp: not(first(part(refl),s)).

Secondly, we use the default rule 'ifthenelse' to declare the regularity that the particle in question immediately precedes the verb, unless when the verb occurs sentence-initially, in which case the particle immediately follows it (which is of course also inexpressible in ID/LP):

```
lp: ifthenelse(first(v(refl),s),
               v(refl) << part(refl),
               part(refl) << v(refl)).
```

These two straightforward LP rules thus are all we need to get exactly the linearizations we want: those of (5a-b) and (7a-b), as well as all and the only other correct expressions derivable from the ID grammar. These LP rules are also interesting in that they express the overall behaviour of a number of other proclitically behaving clitics (as e.g. those with nominal and verbal nature; see above).

Because of space limitations we cannot enter into further details here. Suffice it to say that EFOG was tested successfully in the description of this very complicated domain [2] as well as in some other hard ordering problems in Bulgarian.

5 Conclusion

Logic grammars have generally failed to handle flexible WO in a satisfactory way. We have described a formalism which allows the grammar-writer to express complex WO rules in a language (including discontinuity) in a concise, modular and natural way. EFOG extends the expressive power of the ID/LP format in both allowing complex LP rules and extending their domain of application.

EFOG is based on a previous version of the formalism, called FOG (*Pericliev and Grigorov 1992*), also seeking to overcome the difficulties with the ID/LP format. FOG however looked for different solutions to the problems (e. g. using LP rules attached to each specific ID rule, rather than global ones, which unnecessarily proliferated the LP part of the grammar; or employing flattening rather than having non-contiguous grammar symbols to the same effect). EFOG is also related to FO-TAG (*Becker et al. 1991*) and the HPSG approach (*Engelkamp et al. 1992*, *Oliva 1992*) in extending the domain of applicability of LP rules. A comparisson with these formalisms is beyond the scope of this study; we may only mention here that our inventory of LP relations is larger, and unlike e. g. the latter approach we do not confine to binary branching trees.

[1] This often results in discontinuities (or non-projectivities). For an automated way of discovering and a description of such constructs in Bulgarian, cf. *Pericliev and Ilarionov 1986*, and *Pericliev 1986*.

[2] For the difficulties in handling the adjectival clitics in pure DCG, cf. *Pericliev 1992a*.

References

Becker T., A. Joshi and O. Rambow (1991). Long-distance scrambling and TAG. *Fifth Conference of the EACL*, Berlin, pp. 21-26.

Dahl, V. (1984). More on Gapping Grammars. *Proc. of the Intern. Conf. on 5th Generation Computer Systems*, ICOT, pp. 669-677.

Dahl, V. and H. Abramson (1984). On Gapping Grammars. *Proc. 2nd Intern. Conf. on Logic Programming*, Uppsala, pp. 77-88.

Dahl, V. and H. Abramson (1990). *Logic Grammars*. Springer.

Engelkamp, J., G. Erbach and H. Uszkoreit (1992). Handling linear precedence constraints by unification. *Annual Meeting of the ACL*.

Gazdar, G. and G. Pullum (1981). Subcategorization, constituent order and the notion of "head". M. Moortgat et al. (eds.) *The Scope of Lexical Rules*, Dordrecht, Holland, pp. 107-123.

Gazdar, G., E. Klein, G. Pullum and I. Sag (1985). *Generalized Phrase Structure Grammar*. Harvard, Cambr., Mass.

Hale, K. (1983). Warlpiri and the grammar of non-configurational languages. *Natural Language and Linguistic Theory*, 1, pp. 5-49.

Kashket, M. (1987). A GB-based parser for Warlpiri, a free-word order language. MIT AI Laboratory.

Kay, M. and L. Karttunen (1984). Parsing a free word order language. D. Dowty et al. (eds.) *Natural Language Parsing*. The Cambridge ACL series.

Oliva, K. (1992). Word order constraints in binary branching syntactic structures. University of Saarland Report (appearing also in *COLING'92*).

Pereira, F.C.N. and D.H.D. Warren (1980). Definite Clause Grammars for Natural Language Analysis. *Artificial Intelligence*, v.13, pp. 231-278.

Pericliev, V. (1986). Non-projective con-structions in Bulgarian. *2nd World Congress of Bulgaristics*, Sofia, pp. 271-280 (in Bulgarian).

Pericliev, V. and I. Ilarionov (1986). Testing the projectivity hypothesis. *COLING'86*, Bonn, pp. 56- 58.

Pericliev, V. (1992a). A referent grammar treatment of some problems in the Bulgarian nominal phrase. *Studia Linguistica*, Stockholm, pp. 49-62.

Pericliev, V. (1992b). The ID/LP format: counter-evidence from Bulgarian, (ms).

Pericliev, V. and A. Grigorov (1992). Extending Definite Clause Grammar to handle flexible word order. B. du Boulay et al. (eds.) *Artificial Intelligence V*, North Holland, pp. 161-170.

Pollard C., I. Sag (1987). *Information-Based Syntax and Semantics*. Vol. 1: Fundamentals. CSLI Lecture Notes No. 13, Stanford, CA.

Sag, I. (1987). Grammatical hierarchy and linear precedence. *Syntax and Semantics*, v.20, pp. 303- 339.

Saint-Dizier, P. (1988). Contextual Discon-tinuous Grammars. *Natural Language Understanding and Logic Programming*, II, North Holland, pp. 29-43.

Steele, S. (1981). Word order variation: a typological study. G. Greenberg (ed.) *Universals of Language*, v.4, Stanford.

Uszkoreit, H. (1985). Linear precedence in discontinuous constituents: complex fronting in German. SRI International, Technical Note 371.

Zwicky, A. (1986). Immediate precedence in GPSG. *OSU WPL 32*, pp. 133-138.